Openmobility

Open. & Mobile.

Writing standalone Qt & Python applications for Android

Martin Kolman
Red Hat & Faculty of Informatics, Masaryk University
http://www.modrana.org/om2013
martin.kolman@gmail.com

Overview

- Writing Android applications with Python and Qt
- How it works
- The pydroid project
- Examples
- Acknowledgement

Writing Android applications with Python and Qt

Lets put it all together !

So that:

- **applications can be written entirely in Python**
- **Qt/QML can be used for the GUI**
- **the result is a standalone Google Play compatible APK package**
- **deployment is as easy as possible**
- **all binary components can be recompiled at will**

Writing Android applications with Python and Qt

Writing Android applications with Python and Qt

- Python has been available for Android for ages
- the Qt graphics library has been available for Android since early 2011
- there have been some proof of concepts of Qt-Python bindings working on Android
- and even some proof of concepts of distributing Python + Qt applications

How it works

How “normal” Android applications look like

- written in the Android Java running on the Dalvik VM
- can have a native backend written in C/C++ using the Android NDK
- application packages are distributed as files with the .apk extension
- there is no support for cross-package dependencies

How it works

Necessitas - Qt for Android

- provides Qt libraries compiled for Android 2.2+
- most of Qt functionality is supported, including OpenGL acceleration and Qt Quick
- handles automatic library downloads & updates through the Ministro service

How it works

Python for Android

- provides a Python interpreter for the Python code to run in, the Python-Qt bindings are compiled against it
- there are multiple projects providing Python for Android, for this initiative, the one from the Kivy project was used:
  - https://github.com/kivy/python-for-android

How it works

PySide – Python-Qt bindings

- **make possible to use basically every public Qt API from Python**
- **compiled for Android**
- **build scripts:**
- **https://github.com/M4rtinK/android-pyside-build-scripts**

How it works

Optional – Qt Components

- based on MeeGo Qt Components
- modified to work with Android screen rotation
- compiled for Android
- source available from Gitorious
- https://qt.gitorious.org/~martink/qt-components/martinks-ineans-qt-components/commits/android

How it works

What needs to be in the package

- Python code of the application
- Python & PySide compiled for Android
- Necessitas Java boilerplate & C++ wrapper
  - gets libs from Ministro and creates the window
  - starts the embedded Python interpreter
- optionally – Qt Components
- As a result, the package has about 15 MB :)
- there are multiple ways to trim it down

How it works

How to create the package

- **Necessitas Qt creator**
  - **basically, just press the “run” button**
  - **you can get an example project from here:**

    https://github.com/M4rtinK/android-pyside-example-project

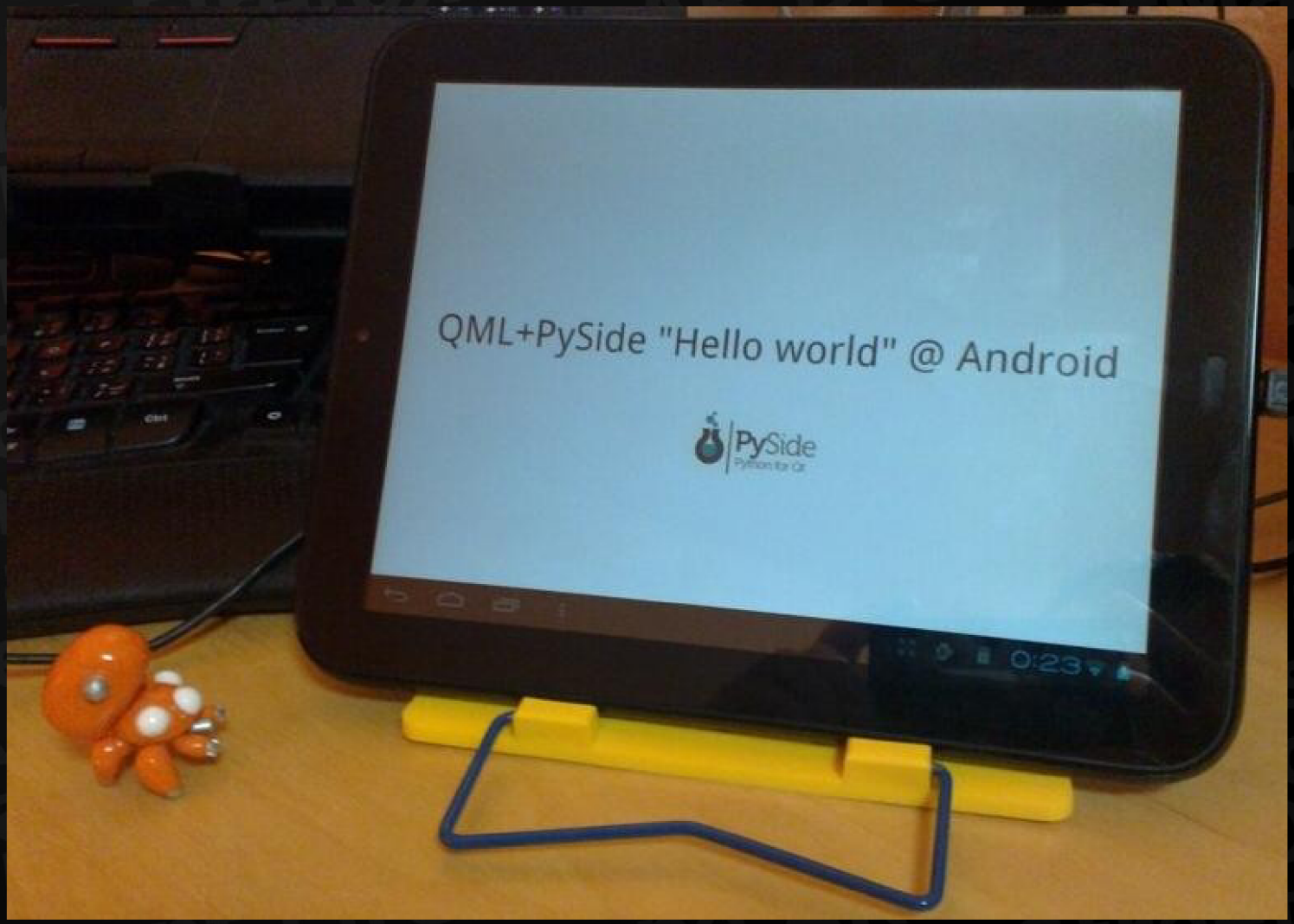
- **packages can be also created from command line using qmake and ant**

How it works

How everything gets in place on first start

- **there are two zip files in the package**
  - **one is for the application, the second for libraries**
- **the boilerplate gets Qt libs from Ministro and then unpacks both zip files folder**
- **then the application is started**
- **all subsequent startups are as fast as for normal Android applications**

How it looks like



Not only Android applications

How it works

The previous image shows

- the Mieru manga and comic book reader
  - Python + PySide + QML + Qt Components
  - https://github.com/M4rtinK/mieru
- running on:
  - Android,
  - BlackBerry 10
  - Nokia N900 & N9
  - PC with Ubuntu 12.10

PySide for Android guide

A detailed guide for building Pyside and using the results to build a Python application APK

- **available on the Qt Project Wiki**

http://qt-project.org/wiki/PySide_for_Android_guide

- **short URL:**

http://bit.ly/Zw6zHf

The pydroid project

is a really nice project thats being developed by Aaron Richiger

- based on my PySide for Android work but even more user/developer friendly
- and MUCH MORE POWERFUL :)
- official website (well, a github repo :) :

  https://github.com/raaron/pydroid
- short URL:

  bit.ly/16Cl3Kj

The pydroid project

features

- efortless setup.py based installation
- automatic project template generation
  - QWidget, QML, QtC, MWC, no-MWC, Pyjnius, etc.
- supports both QtCreator and CLI only for package generation and deployment
- automatically creates the app & lip zip bundles
- progress bar during first start/installation
- supports pip for adding modules to the project

The pydroid project

more features

- support for logging from python directly to the android log
  - that is piped directly to the QtCreator console :)
- fast deployment to an Android device
  - 5 seconds from pressing the “fast deploy” button to application finishing startup on device
- bash autocomplete support :)
- pydroid & project diagnostics

Installing pydroid

clone from Git

git clone https://github.com/raaron/pydroid.git

install

- sudo python setup.py

and restart your shell, or else autocomplete will not work :)

- if you want to use command line deployment, don't forget to fill in the paths ~/.pydroid/deploy.conf

Generate an example application

we will use the Qt Components example

pydroid create example qt_components

- there are also other examples
  - QML only, Qwidget, Pyjnius, etc.

Deploying with Qt Creator

just open the *.pro* file in the Necessitas Qt Creator

- set architecture of your device
- and hit “Run” (ctrl + r)
- and that's it :)

Examples

header

-

Acknowledgements & sources

- Aaron Richiger for the wonderful pydroid project
- THPs PySide for Android – showing that this is possible
- Adrià Cereto-Massagué – integrated & improved THPs patches for Shiboken and PySide
- the Android-Python2.7 project – solved the APK bundling issue
- the Kivy project - provides Android-buildable Python 2.7
- the BlackBerry-Py Building PySide guide – I've used this as a base when making the Android build scripts
- the Necessitas project – made Qt on Android possible
  - also provides the Necessitas Qt Creator used for by the example project for building standalone APKs
- Qt-Project – provides the GUI toolkit :)
- PySide – provides the Python-Qt bindings Ineans Qt Components – with small modifications used in the example application & project